# BAB II

# LANDASAN TEORI

## 2.1 Analisis Sentimen

Analisis sentimen atau *opinion mining* merupakan daerah penelitian perhitungan untuk memahami, mengesktraksi dan mengolah data tekstual secara otomatis untuk mendapatkan informasi sentimen yang terkandung dalam suatu kalimat opini. Analisis sentimen dilakukan untuk melihat pendapat atau kecenderungan kalimat yang sifatnya opini terhadap sebuah masalah atau objek oleh seseorang, apakah cenderung positif atau negatif. Kegiatan klasifikasi sentimen didasarkan pada gagasan bahwa dokumen atau teks mengekspresikan pendapat tentang suatu entitas dari seseorang dan mencoba untuk mengukur sentimen orang tersebut terhadap entitas (Serrano Guerrero dkk., 2015). Analisis sentimen yang dilakukan pada penelitian ini menggunakan tiga *class attribute*, yaitu netral, positif dan negatif.

## 2.2 Kementrian Pekerjaan Umum dan Perumahan Rakyat

Kementrian Pekerjaan Umum dan Perumahan Rakyat Republik Indonesia atau disebut juga dengan Kemen PUPR RI adalah kementrian dalam pemerintah Indonesia yang membidangi urusan pekerjaan umum dan perumahan rakyat. Kementrian PUPR terbentuk pada Tahun 2015 ditandai dengan adanya Peraturan Presiden Republik Indonesia Nomor 15 Tahun 2015 tentang Kementrian Pekerjaan Umum dan Perumahan Rakyat. Kementrian PUPR merupakan salah satu entitas pelaporan yang berupaya konkrit dalam mewujudkan transparansi dan akuntabilitas

pengelolaan keuangan negara dengan menyampaikan laporan pertanggungjawaban keuangan pemerintah yang terdiri dari laporan realisasi anggaran, neraca, laporan operasional, laporan perubahan ekuitas dan catatan atas laporan keuangan (Adha dkk., 2019). Kementrian PUPR memiliki tugas menyelenggarakan urusan pemerintahan di bidang pekerjaan umum dan perumahan rakyat untuk membantu Presiden dalam menyelenggarakan pemerintahan negara.

## 2.3 YouTube API dan Crawling

Dalam proses pengumpulan data komentar, *Application Programming Interface* (API) sendiri telah disediakan oleh YouTube. YouTube API merupakan sebuah dokumentasi yang terdiri dari interface, kelas, fungsi, kumpulan perintah dan juga protokol yang berguna untuk mempermudah *developer* dalam mengakses informasi berupa statistik video dan data saluran YouTube melalui dua jenis panggilan yaitu REST dan XML-RPC. Pada dasarnya fungsi serta perintah pada Youtube API merupakan alat pemanggil *system calls* yang mana berhubungan langsung ke sistem operasinya (Pramana dkk., 2019). Sebelum dapat menggunakan YouTube API, *developer* yang ingin mengakses diwajibkan untuk registrasi terlebih dahulu agar mendapatkan *key* hak akses. *Key* didapatkan ketika *developer* telah berhasil mendaftarkan ke *developer* resmi Youtube.

Crawling data merupakan tahap yang bertujuan untuk mengumpulkan data yang kemudian data tersebut diolah sesuai dengan kebutuhan dari pengguna tersebut. Crawling pada pengumpulan data dilakukan dengan mengunduh data dari server YouTube (Rahardja dkk., 2018).

## 2.4 Text Mining

*Text Mining* atau sering disebut dengan Pemrosesan Teks merupakan salah satu bidang pengetahuan pada *Artificial Intelligence* dan merupakan metode untuk menggali data berupa teks yang dilakukan oleh komputer untuk mendapatkan sesuatu yang baru, dengan tujuannya ialah menemukan suatu informasi yang tersirat dan dapat mewakili isi dari data teks sehingga dapat dilakukan analisa keterhubungan antar data. *Text Mining* mengacu pada pencarian informasi, pertambangan data, *machine learning,* statistik, dan komputasi linguistik terhadap informasi yang disimpan sebagai teks (Bridge, C 2011). *Text Mining* melakukan konversi data teks ke bentuk *semi-structured data. Text Mining* dapat digunakan dalam beberapa hal yaitu ekstraksi informasi, *summarization*, kategorisasi, *topic tracking* dan *clustering*.

Salah satu implementasi dari *text mining* adalah tahap *text preprocessing*. *Text preprocessing* digunakan untuk mengubah data teks yang tidak terstruktur menjadi data yang terstruktur. Tahap *text preprocessing* merupakan proses untuk mempersiapkan data mentah sebelum data masuk ke proses lainnya. Pada proses ini umumnya, data dilakukan dengan cara mengeliminasi data yang tidak sesuai dan mengubah data menjadi bentuk yang lebih mudah diproses oleh sistem. Praproses sangat penting dilakukan dalam analisa sentimen untuk memisahkan komentar negatif dan positif.

## 2.5 Ekstraksi Fitur Bigram dan Term Frequency-Inverse Document Frequency (TF-IDF)

NGram merupakan metode yang diaplikasikan untuk pembangkitan kata atau karakter dengan teknik yang didasarkan pada pemisahan teks menjadi sejumlah *n* karakter dari sebuah *string*. Metode NGram digunakan untuk mengolah teks dan menghitung probabilitas kata setelah kata tertentu. Menurut Rahmawan (2011) NGram model dibagi menjadi 2 kategori yaitu berbasis karakter dan berbasis kata. NGram berbasis karakter menganalisa *string* dari karakter per karakter sedangkan NGram berbasis kata menganalisa *string* dari kata per kata.

Pada penelitian ini, jenis teknik NGram yang digunakan ialah *bigram* dan *trigrams*. *Bigram* merupakan klasifikasi teks dengan menggerakkan dua rangkaian kata maju ke depan menjadi (*n* = 2) secara terurut. Sedangkan *trigrams* merupakan klasifikasi teks dengan menggerakkan tiga kata maju ke depan menjadi (n = 3) secara terurut. NGram yang digunakan ialah berbasis kata dan karakter. Berikut merupakan contoh penerapan NGram basis kata pada kalimat "Analisis sentimen merupakan salah satu teknik dalam mengekstrak informasi berupa pandangan seseorang terhadap suatu kejadian".

Tabel 2.1 Bigram dan Trigrams Basis Kata

| NGram | Hasil Penerapan |
|---|---|
| *Bigram* | Analisis Sentimen, sentimen merupakan, merupakan salah, salah satu, satu teknik, teknik dalam, dalam mengekstrak, mengekstrak informasi, informasi berupa, berupa pandangan, pandangan seseorang, seseorang terhadap, terhadap suatu, suatu kejadian |
| *Trigrams* | Analisis sentimen merupakan, sentimen merupakan salah, merupakan salah satu, salah satu teknik, satu teknik dalam, teknik dalam mengekstrak, dalam mengekstrak informasi, mengekstrak informasi berupa, informasi berupa pandangan, berupa pandangan seseorang, pandangan seseorang terhadap, seseorang terhadap suatu, terhadap suatu kejadian |

Berikut merupakan contoh penerapan Ngram basis karakter pada kalimat "Analisis sentimen".

Tabel 2.2 Bigram dan Trigrams Basis Karakter

| **NGram** | **Hasil Penerapan** |
|---|---|
| *Bigram* | An, na, al, li, is, si, is, s , s, se, en, nt, ti, im, me, en |
| *Trigrams* | Ana, nal, ali, lis, isi, sis, s , s, se, sen, ent, nti, tim, ime, men |

Metode *Term Frequency-Inverse Document Frequency* atau TF-IDF merupakan salah satu fitur dalam proses teks dengan metode integrasi antar *term frequency* (TF) dan *inverse document frequency* (IDF). Menurut Mustaqhfiri (2011), metode Term Frequency-Inverse Document Frequency (TF-IDF) merupakan suatu cara untuk memperoleh pembobotan berdasarkan jumlah kemunculan suatu kata (*term*) dalam sebuah dokumen *term frequency* (TF) dan jumlah kemunculan term dalam koleksi dokumen *inverse document frequency* (IDF) (Mustaqhfiri, 2011).

Perhitungan dimulai dengan menghitung jumlah atau frekuensi kemunculan setiap kata dalam sebuah dokumen. Secara sederhana, metode ini digunakan untuk mengetahui seberapa sering suatu kata muncul dalam dokumen. Menurut Delta Sierra (2019), pada Term Frequency (TF) beberapa jenis formula yang dapat digunakan:

1. TF biner (*binary* TF), hanya memperhatikan apakah suatu kata atau *term* ada atau tidak dalam dokumen, jika ada diberi nilai satu (1), jika tidak diberi nilai nol (0).

2. TF murni (*raw* TF), nilai TF diberikan berdasarkan jumlah kemunculan suatu *term* di dokumen. Contohnya, jika muncul lima (5) kali maka kata tersebut akan bernilai lima (5).
3. TF normalisasi, menggunakan perbandingan antara frekuensi sebuah *term* dengan nilai maksimum dari keseluruhan atau kumpulan frekuensi *term* yang ada pada suatu dokumen. Term frequency dihitung dengan kemunculan term ke-i dalam dokumen ke-j, ditunjukkan pada Persamaan (2.1).

$$TF_{i,j} = \frac{freq_i(d_j)}{\sum_{k=1}^{l} freq_i(d_k)} \tag{2.1}$$

4. TF logaritmik, hal ini untuk menghindari dominansi dokumen yang mengandung sedikit *term* dalam *query*, namun mempunyai frekuensi yang tinggi. Berikut adalah Persamaan (2.2) TF Logaritmik.

$$TF = \begin{cases} 1 + \log_{10}\ (f_{t,d}),\ f_{t,d} > 0 \\ 0, f_{t,d} = 0 \end{cases} \tag{2.2}$$

Metode Inverse Document Frequency (IDF) merupakan sebuah perhitungan dari bagaimana term didistribusikan secara luas pada koleksi dokumen yang bersangkutan (Delta, S 2019). Untuk menentukan nilai IDF diformulasikan pada Persamaan (2.3).

$$IDF_i = \log(\frac{D}{df_i}) \tag{2.3}$$

Keterangan Persamaan 2.3:

$D$ : jumlah semua dokumen

$df_i$ : dokumen yang mengandung term

Rumus umum untuk *Term Weighting* TF-IDF adalah penggabungan dari formula perhitungan *raw* TF dengan formula IDF dengan cara mengalikan nilai TF dengan nilai IDF, Persamaan (2.4) dan Persamaan (2.5):

$$W_{ij} = TF_{ij} * IDF_i \tag{2.4}$$

$$W_{ij} = TF_{ij} * \log(\frac{D}{df_i}) \tag{2.5}$$

Keterangan Persamaan 2.4 dan Persamaan 2.5:

$W_{ij}$ : bobot ( $t_j$ ) terhadap dokumen ( $d_i$ )

$tf_{ij}$ : jumlah kemunculan *term* ( $t_j$ )

$df_i$ : jumlah dokumen yang mengandung *term* ( $t_j$ ) dengan minimal satu kata

## 2.6 Gaussian Naïve Bayes

*Naïve Bayes* merupakan metode klasifikasi yang berakar pada teorema bayes. Algoritma *Naïve Bayes* sebagian besar digunakan dalam analisis sentimen, sistem rekomendasi, dan lainnya. Algoritma ini cenderung cepat dan dapat diperluas ke atribut bernilai nyata, serta yang paling umum dengan mengasumsikan distribusi *Gaussian*. Perpanjangan *Naïve Bayes* disebut Gaussian Naïve Bayes. Fungsi lain dari algoritma ini ialah dapat digunakan untuk memperkirakan distribusi data, tetapi

*Gaussian* merupakan yang paling mudah digunakan karena hanya perlu memperkirakan rata-rata dan standar deviasi dari dataset.

Manfaat dari *Naïve Bayes* dalam bidang pemrosesan teks ialah kita bisa mengetahui penulis dari suatu tulisan hanya dengan mengukur kemiripan pola penggunaan kata pada tulisan tersebut dengan tulisan yang sudah menjadi data latih (Saleh, 2014).

Pada proses dasar *Naïve Bayes*, diasumsikan penyederhanaan nilai atribut bebas bersyarat ketika diberikan nilai *output*. *Naïve Bayes* menghitung probabilitas untuk nilai input pada setiap kelas menggunakan frekuensi. Data latih direpresentasikan dengan set atribut nilai input (x) untuk setiap kelas untuk merangkum distribusi. Setiap nilai *x* memiliki *n* atribut yang direpresentasikan dengan x1, x2, x3, … , xn. Rumus Naïve Bayes dapat dilihat pada Persamaan (2.6).

$$p(C_i|X) = \frac{p(C_i) \ * \ p(X|C_i)}{p(X)} \qquad (2.6)$$

Keterangan Persamaan 2.6:

$p(C_i|X)$ : Probabilitas *posterior*, merupakan peluang dari sebuah hipotesis kelas $C_i$ yang berdasarkan kondisi (bersyarat) data *X*

$p(C_i)$ : *prior*, merupakan sebuah probabilitas dari sebuah hipotesis dalam kelas $C_i$

$p(X|C_i)$ : *likelihood*, merupakan peluang ditemukannya data *X* dikelas $C_i$

$p(X)$ : *evidence* atau marginal *likelihood*, merupakan peluang data *X*

Jika data yang perlu dilatih terdapat lebih dari satu atribut, maka nilai *likelihood* dapat diperoleh dengan menggunakan Persamaan (2.7).

$$p(X|C_i) = \sum_{k=1}^{n} p(x_k|C_i) \qquad (2.7)$$

Keterangan Persamaan 2.7:

*n* : jumlah atribut pada kasus

*k* : indeks untuk menunjukkan nilai atribut *x*

$x_k$ : nilai dari atribut ke *k*

Jika data yang digunakan pada suatu atribut merupakan data numerik kontinu, maka untuk memperoleh nilai *likelihood* dapat menggunakan distribusi *Gaussian* dengan mendistribusikan asumsi nilai kontinu yang diasosiasikan pada setiap kelas. Persamaan (2.8) menunjukkan cara untuk memperoleh *likelihood* pada data kontinu.

$$p(X = v|C_k) = \frac{1}{\sqrt{2\pi\sigma_k^2}}\, e^{-\frac{(v-\mu_k)^2}{2\sigma_k^2}} \qquad (2.8)$$

Keterangan Persamaan 2.8:

$p(X = v|C_k)$ : peluang distribusi ditemukannya nilai *X* dari kelas $C_k$

*v* : sampel data yang diujikan terhadap data training

$\mu_k$ : nilai rata-rata dari *X* yang terasosiasikan dalam setiap kelas dari $C_k$

$\sigma_k^2$ : nilai varians yang menghitung seberapa jauh satu set angka yang tersebar dari nilai rata-rata. Diperoleh dari nilai X yang diasosiasikan oleh kelas $C_k$

$e$ : atau disebut juga $exp$, merupakan nilai exponensial yang ekuivalen dengan 2,7183

$\pi$ : nilai *phi*, setara dengan 3,14

$I$ : indeks yang menunjukkan atribut

$K$ : indeks yang menunjukkan kelas

Cara menghitung nilai rata-rata dari suatu data numerik ditunjukkan pada Persamaan (2.9).

$$\mu = \frac{\sum_{1=1}^{n} x_i}{n} \tag{2.9}$$

Keterangan Persamaan 2.9:

$\mu$ : Nilai rata-rata dari *x* yang terasosiasikan dalam setiap kelas dari $C_k$

$X$ : Sebuah dataset training yang memiliki nilai kontinu. Dalam kasus ini, $X$ merupakan nilai dari rugi daya.

$n$ : Jumlah dati data *x*

Rumus untuk menghitung nilai varians dari suatu data numerik terdapat pada Persamaan (2.10).

$$\sigma^2 = \frac{\sum_{1=1}^{n}(x_i - \mu)^2}{n} \quad (2.10)$$

Keterangan Persamaan 2.10:

$\sigma^2$ : Nilai varians dari populasi

$\mu$ : Nilai rata-rata dari populasi

$x_i$ : Nilai dari data *x* pada indeks ke *i*

*n* : Jumlah dari data *x*

## 2.7 Evaluasi Precision, Recall dan Accuracy

Dalam penelitian ini, pengujian hasil klasifikasi menggunakan pendekatan evaluasi dengan rumus akurasi, yaitu perhitungan untuk menentukan data uji apakah presisi atau benar. Nilai akurasi didapatkan hasil yang diperoleh dengan menggunakan *confusion matrix*. *Confusion matrix* memberikan informasi perbandingan hasil klasifikasi yang dilakukan oleh sistem (model) dengan hasil klasifikasi sebenarnya dalam bentuk tabel matriks. Berikut adalah tabel dari *confusion matrix* dengan 4 kombinasi nilai prediksi dan aktual yang berbeda.

Tabel 2.3 Confusion Matrix

| Prediction | True Values | |
|---|---|---|
| | TRUE | FALSE |
| TRUE | TP<br>Correct Result | FP<br>Unexpected Result |
| FALSE | FN<br>Missing Result | TN<br>Correct Absence of Result |

Keterangan untuk Tabel 2.3 dinyatakan sebagai berikut:

*True Positive* (TP) : Merupakan data positif yang diprediksi dengan benar.

True Negative (TN) : Merupakan data negatif yang diprediksi benar.

False Positive (FP) : Merupakan data negatif namun diprediksi sebagai data positif.

False Negative (FN) : Merupakan data positif namun diprediksi sebagai data negatif.

Pengujian performa yang diukur dilakukan berdasarkan *confusion matrix* dengan ketiga proses, yaitu *precision, recall dan accuracy*. *Precision, recall dan accuracy* merupakan parameter yang digunakan dalam pengujian hasil untuk mengetahui seberapa baik kualitas dari pengujian yang dilakukan. *Precision* dianggap sebagai alat ukur ketepatan atau ketelitian, sedangkan *recall* ialah kesempurnaan. *Precision* merupakan alat ukur tingkat kecocokan antara informasi yang dicari oleh pengguna dengan hasil yang diberikan oleh sistem. *Recall* merupakan alat ukur tingkat kesuksesan sistem dalam mendapatkan informasi kembali. *Accuracy* merupakan proporsi jumlah prediksi dataset yang benar. Satuan dari nilai *precision*, *recall* dan *accuracy* yang digunakan adalah persen. Hasil kesimpulan didapatkan dari perhitungan f-score. F-score merupakan nilai yang mengkombinasikan nilai precision dan recall (Steinberger dan Ježek, 2009). F-score bernilai 1 maka dapat dikatakan telah memiliki nilai maksimal dengan kualitas sangat baik, sebaliknya jika bernilai 0 maka f-score belum dianggap baik. Hal ini ditentukan dengan menggunakan Persamaan (2.11), Persamaan (2.12), dan Persamaan (2.13).

$$Precision = \frac{TP}{TP + FP} \quad (2.11)$$

$$Recall = \frac{TP}{TP + FN} \quad (2.12)$$

$$Accuracy = \frac{TP\ +\ TN}{TP\ +\ TN +\ FP\ +\ FN} * 100\% \quad (2.13)$$

Keterangan Persamaan 2.11, Persamaan 2.12, dan Persamaan 2.13:

| | |
|---|---|
| $TF$ | : jumlah prediksi yang tepat bersifat positif |
| $TF + FP$ | : jumlah dokumen yang dipilih sistem sebagai ringkasan |
| TP + FN | : jumlah dokumen yang dipilih pakar sebagai ringkasan |

Untuk rumus f-score terdapat pada persamaan (2.14).

$$f - score = \frac{2\ Precision\ Recall}{Precision + Recall} \quad (2.14)$$